

On the Measure of the Conflicts: A MUS-Decomposition Based Framework

Said Jabbour¹, Yue Ma², Badran Raddaoui¹, Lakhdar Saïs¹, Yakoub Salhi¹

¹CRIL - CNRS, Université d'Artois, France
{jabbour, raddaoui, saïs, salhi}@cril.fr

² Technische Universität Dresden, Institut für theoretische Informatik, Dresden, Germany
mayue@tcs.inf.tu-dresden.de

Abstract

Measuring inconsistency is viewed as an important issue related to handling inconsistencies. Good measures are supposed to satisfy a set of rational properties. However, defining sound properties is sometimes problematic. In this paper, we emphasize one such property, named *Decomposability*, rarely discussed in the literature due to its modeling difficulties. To this end, we propose an independent decomposition which is more intuitive than existing proposals. To analyze inconsistency in a more fine-grained way, we introduce a graph representation of a knowledge base and various MUS-decompositions. One particular MUS-decomposition, named *distributable MUS-decomposition* leads to an interesting partition of inconsistencies in a knowledge base such that multiple experts can check inconsistencies in parallel, which is impossible under existing measures. Such particular *MUS-decomposition* results in an inconsistency measure that satisfies a number of desired properties. Moreover, we give an upper bound complexity of the measure that can be computed using 0/1 linear programming or Min Cost Satisfiability problems, and conduct preliminary experiments to show its feasibility.

1 Introduction

Conflicting information is often unavoidable for large-sized knowledge bases (KBs for short). Thus, analyzing conflicts has gained a considerable attention in Artificial Intelligence research (Bertossi, Hunter, and Schaub 2005). In the same vein, measuring inconsistency has proved useful and attractive in diverse scenarios, including software specifications (Martinez, Arias, and and 2004), e-commerce protocols (Chen, Zhang, and Zhang 2004), belief merging (Qi, Liu, and Bell 2005), news reports (Hunter 2006), integrity constraints (Grant and Hunter 2006), requirements engineering (Martinez, Arias, and and 2004), databases (Martinez et al. 2007; Grant and Hunter 2013), semantic web (Zhou et al. 2009), and network intrusion detection (McAreavey et al. 2011).

Inconsistency measuring is helpful to compare different knowledge bases and to evaluate their quality (Grant 1978). A number of logic-based inconsistency measures have been studied, including the maximal η -consistency

(Knight 2002), measures based on variables or via multi-valued models (Grant 1978; Hunter 2002; Oller 2004; Hunter 2006; Grant and Hunter 2008; Ma et al. 2010; Xiao et al. 2010; Ma, Qi, and Hitzler 2011), n -consistency and n -probability (Doder et al. 2010), minimal inconsistent subsets based inconsistency measures (Hunter and Konieczny 2008; Mu, Liu, and Jin 2011; Mu, Liu, and Jin 2012; Xiao and Ma 2012), Shapley inconsistency value (Hunter and Konieczny 2006a; Hunter and Konieczny 2010), and more recently the inconsistency measurement based on minimal proofs (Jabbour and Raddaoui 2013).

There are different ways to categorize the proposed measures. One way is with respect to their dependence on syntax or semantics: Semantic based ones aim to compute the proportion of the language that is affected by the inconsistency, via for example paraconsistent semantics. Whilst, syntax based ones are concerned with the minimal number of formulae that cause inconsistencies, often through minimal inconsistent subsets. Different measures can also be classified by being formula or knowledge base oriented. For example, the inconsistency measures in (Hunter and Konieczny 2006a; Hunter and Konieczny 2010) consist in quantifying the contribution of a formula to the inconsistency of a whole knowledge base containing it, while the other mentioned measures aim to quantify the inconsistency degree of the whole knowledge base. Some basic properties (Hunter and Konieczny 2010) such as *Consistency*, *Monotony*, *Free Formula Independence*, are also proposed to evaluate the quality of inconsistency measures.

In this paper, we propose a syntax-based framework to measure inconsistencies¹ using a novel methodology allowing to resolve inconsistencies in a parallel way. To this end, *distributable MUS-decomposition* and *distribution index* of a KB are introduced. Intuitively, a distributable MUS-decomposition gives a reasonable partition of a KB such that it allows multiple experts to solve inconsistencies in parallel; And the distribution index is the maximal components that a KB can be partitioned into. This methodology is of great importance in a scenario where the information in a KB is precious, large, and complex such that removing or weakening information requires intensive and time-consuming interac-

¹It can be embedded into Shapley Inconsistency Value to have a formula oriented measure (Hunter and Konieczny 2006a).

tions with human experts. Consider $K = \{a_1, \neg a_1, a_1 \vee \neg a_2, a_2, \neg a_2, \dots, a_{n-1}, \neg a_{n-1}, a_{n-1} \vee \neg a_n, a_n, \neg a_n\}$. Intuitively, K contains a large number of inconsistencies. And interestingly, our approach can recognize $\{a_i, \neg a_i\}$ as n distributable parts of K such that each expert can focus on verifying a single part carefully and independently². In contrast, classical approaches follow the idea of resolving inconsistency as a whole without being able to break a KB into independent pieces. Take, for example, the classical Hitting Set approach which identifies a minimal set of formulae, e.g. $\{\neg a_i \mid 1 \leq i \leq n\}$ of K , to remove for restoring consistency. Note that K has many such Hitting Sets of a big size n . Therefore, even if working in parallel, each expert needs to verify a large number of formulae, which is time consuming. More problematic in general, there are often overlaps among Hitting sets so that multiple experts have to waste time in unnecessarily rechecking the overlaps. This is the same if we simply distribute one minimal inconsistent subsets to an expert. However, the proposed distributable MUS-decomposition avoids this problem because it gives a disjoint decomposition of a KB. The methodology is inspired and a side-product of our exploration of the *decomposition* property defined for inconsistency measures, which is rarely discussed in the literature due to its modeling difficulty (Hunter and Konieczny 2010).

Our technical contributions are as follows:

- We propose *independent decomposability* as a more reasonable characterization of inconsistency measures.
- We define a *graph representation* of KBs to analyze connections between minimal inconsistent subsets by exploiting the structure of the graph. Such a representation is then used to improve an existing inconsistency measure to satisfy the independent decomposability.
- Based on the graph representation, a series of *MUS-decompositions* are introduced and used for defining the *distribution-based inconsistency measure* I_D . We show the interesting properties of I_D and give a comparison with other measures, which indicates its rationality.
- We study the complexity of I_D (via an *extended set packing problem*) and we provide encodings as a 0/1 linear program or min cost satisfiability for its computation.

The paper is organized as follows: Sections 2 and 3 give basis notions and recall some inconsistency measures relevant to the present work. In Section 4, we propose a graph representation of a KB and use it to revise an existing measure. Section 5 focuses on *MUS-decomposition* and *distribution-based inconsistency measure*. Section 6 gives the complexity results of the proposed measure and its computation algorithms whose efficiency is evaluated in Section 7. Section 8 concludes the paper with some perspectives.

2 Preliminaries

Through this paper, we consider the propositional language \mathcal{L} built over a finite set of propositional symbols \mathcal{P} using

²More details are explained later in the paper.

classical logical connectives $\{\neg, \wedge, \vee, \rightarrow\}$. We will use letters such as a and b to denote propositional variables, Greek letters like α and β to denote propositional formulae. The symbols \top and \perp denote tautology and contradiction, respectively.

A knowledge base K consists of a finite set of propositional formulae. Sometimes, a propositional formula can be in conjunctive normal form (CNF) i.e. a conjunction of clauses. Where a clause is a disjunction literals, and a literal is either a propositional variable (x) or its negation ($\neg x$). For a set S , $|S|$ denotes its cardinality. Moreover, a KB K is inconsistent if there is a formula α such that $K \vdash \alpha$ and $K \vdash \neg \alpha$, where \vdash is the deduction in classical propositional logic. If K is inconsistent, *Minimal Unsatisfiable Subsets* (MUS) of K are defined as follows:

Definition 1 (MUS). *Let K be a KB and $M \subseteq K$. M is a minimal unsatisfiable (inconsistent) subset (MUS) of K iff $M \vdash \perp$ and $\forall M' \subsetneq M$, $M' \not\vdash \perp$. The set of all minimal unsatisfiable subsets of K is denoted $MUSes(K)$.*

Clearly, an inconsistent KB K can have multiple minimal inconsistent subsets. When a MUS is singleton, the single formula in it, is called a *self-contradictory formula*. We denote the set of self-contradictory formulae of K by $selfC(K) = \{\alpha \in K \mid \{\alpha\} \vdash \perp\}$. A formula α that is not involved in any MUS of K is called *free formula*. The set of free formulae of K is written $free(K) = \{\alpha \mid \text{there is no } M \in MUSes(K) \text{ such that } \alpha \in M\}$, and its complement is named *unfree formulae* set, defined as $unfree(K) = K \setminus free(K)$. Moreover, the *Maximal Consistent Subset* and *Hitting set* are defined as follows:

Definition 2 (MSS). *Let K be a KB and M be a subset of K . M is a maximal satisfiable (consistent) subset (MSS) of K iff $M \not\vdash \perp$ and $\forall \alpha \in K \setminus M$, $M \cup \{\alpha\} \vdash \perp$. The set of all maximal satisfiable subsets is denoted $MSSes(K)$.*

Definition 3. *Given a universe U of elements and a collection \mathcal{S} of subsets of U , $H \subseteq U$ is a hitting set of \mathcal{S} if $\forall E \in \mathcal{S}$, $H \cap E \neq \emptyset$. H is a minimal hitting set of \mathcal{S} if H is a hitting set of \mathcal{S} and each $H' \subset H$ is not a hitting set of \mathcal{S} .*

3 Inconsistency Measures

We review the inconsistency measures relevant to the ones proposed in this paper.

There have been several contributions for measuring inconsistency in knowledge bases defined through minimal inconsistent subsets theories. In (Hunter and Konieczny 2010), Hunter and Konieczny introduce a scoring function allowing to measure the degree of inconsistency of a subset of formulae of a given knowledge base. In other words, for a subset $K' \subseteq K$, the scoring function is defined as the reduction of the number of minimal inconsistent subsets obtained by removing K' from K (i.e. $|MUSes(K)| - |MUSes(K - K')|$). By extending the scoring function, the authors introduce an inconsistency measure of the whole base, defined as the number of minimal inconsistent subsets of K . Formally, $I_{MI}(K) = |MUSes(K)|$.

I_{MI} measure also leads to an interesting Shapley Inconsistency Value $S_{\alpha}^{I_{MI}}$ with desirable properties (Hunter and Konieczny 2010).

Combining both minimal inconsistent subsets and maximal consistent subsets is another way to define inconsistency degree (Mu et al. 2011; Grant and Hunter 2011). We consider the inconsistency value $I_M(K)$ that counts for a given KB, the number of its $MSSes$ and its Self-contradictory formulae (subtraction of 1 is required to make $I_M(K) = 0$ when K is consistent):

$$I_M(K) = |MSSes(K)| + |selfC(K)| - 1.$$

Another inconsistency measure considered in this paper is defined as the minimum hitting set of $MUSes(K)$:

$$\delta_{hs}(K) = \min\{|H| \mid H \text{ is a hitting set of } MUSes(K)\}.$$

$\delta_{hs}(K)$ is the size of the smallest hitting set of $MUSes(K)$ w.r.t. its cardinality.

In addition, a set of properties have been proposed to characterize an inconsistency measure.

Definition 4 ((Hunter and Konieczny 2010)). *Given two knowledge bases K and K' , and formulae α and β in \mathcal{L} ,*

- (1) *Consistency:* $I(K) = 0$ iff K is consistent
- (2) *Monotony:* $I(K) \leq I(K \cup K')$
- (3) *Free Formula Independence:* if α is a free formula in $K \cup \{\alpha\}$, then $I(K \cup \{\alpha\}) = I(K)$
- (4) *MinInc:* If $M \in MUSes(K)$, then $I(M) = 1$.

The monotony property shows that the inconsistency value of a KB increases with the addition of new formulae. The free formula independence property states that the set of formulae not involved in any minimal inconsistent subset does not influence the inconsistency measure. The MinInc is used to characterize the Shapley Inconsistency Value by $I_{MI}(K)$ in (Hunter and Konieczny 2008).

4 Independent Decomposability Property

There are common properties that we examine for an inconsistency measure (Definition 4), while leaving another property, called *Decomposability* or *Additivity*, debatable due to its modelling difficulty (Hunter and Konieczny 2008). Indeed, properties in Definition 4 have an inspiring root from the axioms of Shapley Value (Shapley 1953). As mentioned in (Luce and Raiffa 1957), one of the main limitation of the original additivity lies in the fact that the interactions of sub-games are not considered. Moreover, (Hunter and Konieczny 2006b) argue that a direct translation of Shapley's additivity has little sense for inconsistency measures. For this reason, *Pre-Decomposability* and *Decomposability* are defined (Hunter and Konieczny 2010) for formula-oriented inconsistency measures.

In this section, we analyze the limitation of existing decomposability property and propose an *Independent Decomposability* which is more intuitive. We then derive a new measure I'_M by modifying I_M to satisfy the independent decomposability property by considering the interactions between $MUSes$ through *MUS-graph representation* of a KB.

Let us recall Pre-decomposability and Decomposability properties (Hunter and Konieczny 2010).

Definition 5 (Pre-Decomposability³). *Let K_1, \dots, K_n be knowledge bases and I an inconsistency measure. I satisfies Pre-Decomposability if it satisfies the following condition: If $MUSes(K_1 \cup \dots \cup K_n) = MUSes(K_1) \oplus \dots \oplus MUSes(K_n)$ ⁴, then $I(K_1 \cup \dots \cup K_n) = I(K_1) + \dots + I(K_n)$.*

Pre-Decomposability ensures that the inconsistency degree of a KB K can be obtained by summing up the degrees of its sub-bases K_i under the condition that $\{MUSes(K_i) \mid 1 \leq i \leq n\}$ is a partition of $MUSes(K)$.

Definition 6 (Decomposability). *I satisfies Decomposability if it satisfies the following condition: If $|MUSes(K_1 \cup \dots \cup K_n)| = \sum_{1 \leq i \leq n} |MUSes(K_i)|$, then $I(K_1 \cup \dots \cup K_n) = I(K_1) + \dots + I(K_n)$.*

Compared to Pre-Decomposability, Decomposability characterizes a weaker condition that consider only $MUSes$ cardinalities of K and K_i . Although Pre-Decomposability and Decomposability can characterize some kind of interactions. We argue that this condition is not sufficient. Let us consider the following example:

Example 1. Let $K_1 = \{a, \neg a\}$, $K_2 = \{\neg a, a \wedge b\}$, $K_3 = \{c, \neg c\}$, each of which contains only one single *MUS*. Consider two bases $K = K_1 \cup K_2$, $K' = K_1 \cup K_3$. Clearly, $MUSes(K) = MUSes(K_1) \oplus MUSes(K_2)$, and $MUSes(K') = MUSes(K_1) \oplus MUSes(K_3)$. For any measure I , if I satisfies the decomposability property (Definition 6), we have $I(K) = I(K_1) + I(K_2)$ and $I(K') = I(K_1) + I(K_3)$. Moreover, if I satisfies the MinInc property. Then, K and K' will have the same value, which is counter-intuitive because the components of $MUSes(K') = \{\{a, \neg a\}, \{c, \neg c\}\}$ are unrelated, whereas those of $MUSes(K) = \{\{a, \neg a\}, \{\neg a, a \wedge b\}\}$ are overlapping. Consequently, the components of $MUSes(K')$ are more spread than those of $MUSes(K)$. One can expect that K' should contain more inconsistencies than K .

This example illustrates the necessity to characterize the interactions among sub-bases whose inconsistency measures can be summed up. To this end, we propose the following independent decomposability property:

Definition 7 (Independent Decomposability). *Let K_1, \dots, K_n be knowledge bases and I an inconsistency measure. If $MUSes(K_1 \cup \dots \cup K_n) = MUSes(K_1) \oplus \dots \oplus MUSes(K_n)$ and $unfree(K_i) \cap unfree(K_j) = \emptyset$ for all $1 \leq i \neq j \leq n$, then $I(K_1 \cup \dots \cup K_n) = I(K_1) + \dots + I(K_n)$. I is then called ind-decomposable.*

To perform additivity for a given measure, the independent decomposability requires an additional precondition expressing that pairwise sub-bases should not share unfree formulae, which encodes a stronger independence among sub-bases. Indeed, the independent decomposability avoids the counter-intuitive conclusion illustrated in Example 1. To

³It is named MinInc Separability in (Hunter and Konieczny 2008)

⁴We denote a partition $\{A, B\}$ of a set C by $C = A \oplus B$, i.e., $C = A \cup B$ and $A \cap B = \emptyset$.

illustrate this, suppose that I satisfies independent decomposability, then we have $I(K') = I(K_1) + I(K_3)$, but not necessarily $I(K') = I(K_1) + I(K_2)$ as $MUSes(K_1)$ and $MUSes(K_2)$ share the formula $\neg a$. Hence $I(K)$ can be different from $I(K')$.

Clearly, the following relations hold among different decomposability conditions.

Proposition 1. *Decomposability implies Pre-Decomposability; Pre-Decomposability implies Independent Decomposability.*

Indeed, as shown by Example 1, the strong constraints of Pre-Decomposability and Decomposability would make an inconsistency measure behavior counter-intuitive. In contrast, the independence between sub-bases required in the independent decomposability property make it more intuitive.

While we can see that the measure I_{MI} is pre-decomposable, decomposable, and ind-decomposable, it is not the case for I_M measure as shown below.

Proposition 2. *The measure I_M is not pre-decomposable, neither decomposable and nor ind-decomposable.*

Proof. Consider the counter example: $K_1 = \{a, \neg a\}$, $K_2 = \{b, \neg b\}$ and $K = K_1 \cup K_2$. It is easy to check that K and K_i ($i = 1, 2$) satisfy the conditions of Pre-Decomposability, Decomposability, and Independent Decomposability. We have $I_M(K_1 \cup K_2) = 3$ while $I_M(K_1) + I_M(K_2) = 2$. Consequently, $I_M(K_1) + I_M(K_2) \neq I_M(K_1 \cup K_2)$. Thus, I_M is not pre-decomposable, neither decomposable and nor ind-decomposable. \square

Indeed, the following theorem states that under certain constraints, MSS is multiplicative instead of additive.

Theorem 3. *Let $K = K_1 \cup \dots \cup K_n$ be KBs such that $MUSes(K_1 \cup \dots \cup K_n) = MUSes(K_1) \oplus \dots \oplus MUSes(K_n)$ and, for all $1 \leq i, j \leq n$ with $i \neq j$, $K_i \cap K_j = \emptyset$. Then, $M \in MSSes(K)$ iff $M = M_1 \cup \dots \cup M_n$ where $M_1 \in MSSes(K_1), \dots, M_n \in MSSes(K_n)$.*

Proof. By induction on n . The case of $n = 1$ is trivial. We now consider the case of $n > 1$. Let $K' = K_1 \cup \dots \cup K_{n-1}$. Using induction hypothesis, we have $M' \in MSSes(K')$ iff $M' = M_1 \cup \dots \cup M_{n-1}$ where $M_1 \in MSSes(K_1), \dots, M_{n-1} \in MSSes(K_{n-1})$.

Part \Rightarrow . Let $M \in MSSes(K' \cup K_n)$. Then, there exist $M' \subseteq K'$ and $M_n \subseteq K_n$ such that $M = M' \cup M_n$. If $M' \notin MSSes(K')$ (resp. $M_n \notin MSSes(K_n)$) then there exists $\alpha \in (K' \cup K_n) \setminus M$ such that $M' \cup \{\alpha\}$ (resp. $M_n \cup \{\alpha\}$) is consistent. Using $MUSes(K' \cup K_n) = MUSes(K') \oplus MUSes(K_n)$ and $K' \cap K_n = \emptyset$, $M \cup \{\alpha\}$ is consistent and we get a contradiction. Therefore, $M' \in MSSes(K')$ and $M_n \in MSSes(K_n)$.

Part \Leftarrow . Let $M' \in MSSes(K')$ and $M_n \in MSSes(K_n)$. Then, the set $M = M' \cup M_n$ is consistent, since we have $M' \cap M_n = \emptyset$ and $MUSes(K' \cup K_n) = MUSes(K') \oplus MUSes(K_n)$. Let us now show that M is in $MSSes(K' \cup K_n)$. Assume that M is not in $MSSes(K' \cup K_n)$. Then, there exists $\alpha \in (K' \cup K_n) \setminus M$ such that $M \cup \{\alpha\}$ is consistent. If $\alpha \in K'$ (resp. $\alpha \in K_n$), then $M' \cup \{\alpha\}$ (resp.

$M_n \cup \{\alpha\}$) is consistent and we get a contradiction. Therefore, M is in $MSSes(K' \cup K_n)$. \square

Using this theorem, we deduce the following corollary:

Corollary 4. *Let $K = K_1 \cup \dots \cup K_n$ be KBs such that $MUSes(K_1 \cup \dots \cup K_n) = MUSes(K_1) \oplus \dots \oplus MUSes(K_n)$ and, for all $1 \leq i, j \leq n$ with $i \neq j$, $K_i \cap K_j = \emptyset$. Then, $|MSSes(K)| = |MSSes(K_1)| \times \dots \times |MSSes(K_n)|$.*

As the Independent Decomposability gives a more intuitive characterization of the interaction among subsets, in the following, we are interested in restoring the independent decomposability property of the I_M measure.

Let us first define two fundamental concepts: *MUS-graph* and *MUS-decomposition*.

Definition 8 (MUS-graph). *The MUS-graph of K of a KB K , denoted $\mathcal{G}_{MUS}(K)$, is an undirected graph where:*

- $MUSes(K)$ is the set of vertices; and
- $\forall M, M' \in MUSes(K)$, $\{M, M'\}$ is an edge iff $M \cap M' \neq \emptyset$.

A MUS-graph of K gives us a structural representation of the connection between minimal unsatisfiable subsets.

Example 2. *Let $K = \{a \wedge d, \neg a, \neg b, b \vee \neg c, \neg c \wedge d, \neg c \vee e, c, \neg e, e \wedge d\}$. We have $MUSes(K) = \{M_1, \dots, M_5\}$ where $M_1 = \{\neg a, a \wedge d\}$, $M_2 = \{c, \neg b, b \vee \neg c\}$, $M_3 = \{c, \neg c \wedge d\}$, $M_4 = \{\neg c \vee e, c, \neg e\}$, and $M_5 = \{\neg e, e \wedge d\}$. So $\mathcal{G}_{MUS}(K)$ is as follows:*

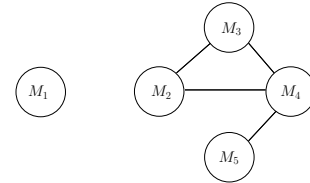


Fig 1: $\mathcal{G}_{MUS}(K)$: MUS-graph of K

Moreover, $\mathcal{G}_{MUS}(K)$ leads to a partition of a KB K , named *MUS-decomposition*, as defined below.

Definition 9 (MUS-decomposition). *A MUS-decomposition of K is a set $\{K_1, \dots, K_p\}$ such that $K = K_1 \cup \dots \cup K_p \cup free(K)$ and $MUSes(K_i)$ ($1 \leq i \leq p$) are the connected components of $\mathcal{G}_{MUS}(K)$.*

By the fact that $MUSes(K) \neq \emptyset$ and the uniqueness of the connected components of a graph, we can easily see:

Proposition 5. *MUS-decomposition exists and is unique for an inconsistent KB.*

Example 3. (Example 2 contd.) *The MUS-decomposition of K contains two components of $\mathcal{G}_{MUS}(K)$: $K_1 = M_1$ and $K_2 = M_2 \cup M_3 \cup M_4 \cup M_5$ by noting that $free(K) = \emptyset$.*

Obviously, the MUS-decomposition of a KB can be computed in polynomial time given its MUS-graph. Interestingly, we can see that the partition $\{K_1, \dots, K_p, free(K)\}$

satisfies the application conditions of Independent Decomposability. That is, if an inconsistency measure I is ind-decomposable and free-formula independent, then $I(K) = I(K_1) + \dots + I(K_p)$.

In the following, based on MUS-decomposition, we present an alternative to the inconsistency measure I_M (defined in Section 2) so as to make it ind-decomposable.

Definition 10. Let K be a KB with its MUS-decomposition $K = \{K_1, \dots, K_p\}$. The I'_M measure is defined as follows:

$$I'_M(K) = \begin{cases} \sum_{1 \leq i \leq p} |MSSes(K_i)| + |selfC(K)| & \text{if } K \vdash \perp; \\ 0 & \text{otherwise.} \end{cases}$$

That is, instead of $MSSes(K)$ as in I_M , the maximal consistent subsets of MUS-decomposition of K_i are used in I'_M .

Example 4. (Example 2 contd.) We have $MSSes(K_1) = \{\{a \wedge d\}, \{\neg a\}\}$ and $MSSes(K_2) = \{\{\neg b, b \vee \neg c, \neg c \wedge d, \neg c \vee e, e \wedge d\}, \{\neg b, b \vee \neg c, \neg c \wedge d, \neg c \vee e, \neg e\}, \{b \vee \neg c, \neg c \vee e, c, e \wedge d\}, \{\neg b, \neg c \vee e, c, e \wedge d\}, \{b \vee \neg c, c, \neg e\}, \{\neg b, c, \neg e\}\}$. Then $I'_M(K) = 2 + 6 = 8$.

Proposition 6. I'_M measure is ind-decomposable.

Proof. Let $K = \bigcup_{1 \leq i \leq n} K_i$ be a KB such that

$MUSes(K) = \bigoplus_{1 \leq i \leq n} MUSes(K_i)$ and, for all $1 \leq i, j \leq n$ with $i \neq j$, $unfree(K_i) \cap unfree(K_j) = \emptyset$. One can easily see that $I'_M(K) = 0$ if and only if, for all $1 \leq i \leq n$, $I'_M(K_i) = 0$. We now consider the case of $I'_M(K) > 0$. We denote by $\mathcal{C}(K_i)$ the set of connected components in $\mathcal{G}_{MUS}(K_i)$ for $i = 1, \dots, n$. Thus, $\bigcup_{1 \leq i \leq n} \mathcal{C}(K_i)$ is the set of connected components in

$\mathcal{G}_{MUS}(K)$, since $\mathcal{G}_{MUS}(K) = \biguplus_{1 \leq i \leq n} \mathcal{G}_{MUS}(K_i)$ ⁵. Moreover, it is obvious that $selfC(K) = \bigcup_{1 \leq i \leq n} selfC(K_i)$.

Let $\{K_i^1, \dots, K_i^{p_i}, free(K_i)\}$ be the MUS-decomposition of K_i for $i = 1, \dots, n$. We have $I'_M(K) = \sum_{1 \leq i \leq n} (\sum_{1 \leq j \leq p_i} |MSSes(K_i^j)| + |selfC(K_i)|) = \sum_{1 \leq i \leq n} I'_M(K_i)$, since $(\bigcup_{1 \leq i \leq n} \{K_i^1, \dots, K_i^{p_i}\}) \cup (\bigcup_{1 \leq i \leq n} free(K_i))$ is the MUS-decomposition of K . \square

That is, by taking into account the connections between minimal inconsistent subsets, MUS-decomposition gives us a way to define an inconsistency measure which still satisfies the Independent Decomposability.

5 A New MUS-based Inconsistency Measure

Recall that we want to have a way to resolve inconsistencies in a parallel way as mentioned in Section 1. Indeed, MUS-decomposition defines a disjoint

⁵The union of disjoint graphs is the graph with the union of vertex and edge sets from individual graphs as its vertexes and edges.

partitions of a KB. However, it is inadequate for this purpose. Consider again $K = \{a_1, \neg a_1, a_1 \vee \neg a_2, a_2, \neg a_2, \dots, a_{n-1}, \neg a_{n-1}, a_{n-1} \vee \neg a_n, a_n, \neg a_n\}$. The MUS-decomposition can not divide K into smaller pieces because its MUS-graph contains only one connected component. A solution to this problem is via a more fine-grained analysis of a MUS-graph by taking into account its inner structures. To this end, we propose *partial* and *distributable* MUS-decompositions, based on which a new inconsistency measure is proposed and shown having more interesting properties.

Let us first study a general characterization of inconsistency measures with respect to the Independent Decomposability property.

Definition 11. Let K be a KB, $\{K_1, \dots, K_p\}$ the MUS-decomposition of K and δ a function from $\{K_1, \dots, K_p\}$ to \mathbb{R} . The MUS-decomposition based inconsistency measure of K with respect to δ , denoted $I_D^\delta(K)$, is defined as follows:

$$I_D^\delta(K) = \sum_{i=1}^p \delta(K_i)$$

A range of possible measures can be defined using the above general definition. Let us review some existing instances of I_D^δ according to some δ functions. The simplest one is obtained when $\delta(K_i) = 1$. In this case, we get a measure that assigns to K the number of its connected components. However, this measure is not monotonic. Indeed, adding new formulae to a KB can decrease the number of connected components. For instance, consider the KB $K = \{a, \neg a, b, \neg b\}$ that contains two singleton connected components $K_1 = \{a, \neg a\}$ and $K_2 = \{b, \neg b\}$. Now, adding the formula $a \vee b$ to K leads to a new KB containing a unique connected component $K = \{a, \neg a, \{b, \neg b\}, \{a, a \vee b, \neg b\}\}$. Besides, this simple measure considers each connected component as an inseparable entity.

Moreover, when we take $\delta(K_i) = |K_i|$ (the number of MUSes involved in the connected component K_i), $I_D^\delta(K)$ is equal to I_{MI} measure i.e. $I_D^\delta(K) = |MUSes(K)|$. This measure again does not take into account the inner structure of minimal inconsistent subsets of a K .

5.1 (Maximal) Partial MUS-decomposition

We now modify I_D^δ to take into account interactions between MUSes. In particular, we deeply explore the Independent Decomposability and the Monotony properties to define a new inconsistency measure, while keeping other desired properties satisfied. To this end, we first introduce the partial MUS-decomposition notion.

Definition 12 (Partial MUS-decomposition). Let K be a KB and K_1, \dots, K_n subsets of K . The set $\{K_1, \dots, K_n\}$ is called a *partial MUS-decomposition* of K if the following conditions are satisfied:

- (1) $K_i \vdash \perp$, for $1 \leq i \leq n$;
- (2) $MUSes(K_1 \cup \dots \cup K_n) = \bigoplus_{1 \leq i \leq n} MUSes(K_i)$;
- (3) $K_i \cap K_j = \emptyset$, $\forall i \neq j$.

We denote $pMUSd(K)$ the set of partial MUS-decompositions of K .

The following proposition comes from the fact that the MUS-decomposition of a KB K is in $pMUSd(K)$.

Proposition 7. *Any inconsistent KB has at least one partial MUS-decomposition.*

Unlike the uniqueness of MUS-decomposition, a KB can have multiple partial MUS-decompositions as shown in the following example.

Example 5. Consider $K = \{a, \neg a, a \vee b, \neg b, b, c, \neg c \wedge d, \neg d \wedge e \wedge f, \neg e, \neg f\}$. Figure 2 depicts the graph representation of K which contains two connected components C_1 and C_2 where $C_1 = \{a, \neg a, a \vee b, \neg b, b\}$ and $C_2 = \{c, \neg c \wedge d, \neg d \wedge e \wedge f, \neg e, \neg f\}$. So the MUS-decomposition of K is $\{C_1, C_2\}$. However, there are many partial MUS-decompositions with some examples listed below:

- $K_1 = \{a, \neg a\}$, and $K_2 = \{b, \neg b\}$.
- $K'_1 = \{a, \neg a\}$, $K'_2 = \{b, \neg b\}$, and $K'_3 = \{c, \neg c \wedge d\}$.
- $K''_1 = \{\neg a, a \vee b, \neg b\}$, and $K''_2 = \{\neg c \wedge d, \neg d \wedge e \wedge f\}$.

Note that $K'_3 = \{c, \neg c \wedge d\}$ and $K''_3 = \{\neg e, \neg d \wedge e \wedge f\}$ can not form a partial MUS-decomposition due to the violation of the condition (2) in Definition 12. This also shows that condition (3) alone can not guarantee to satisfy the condition 2 in the definition.

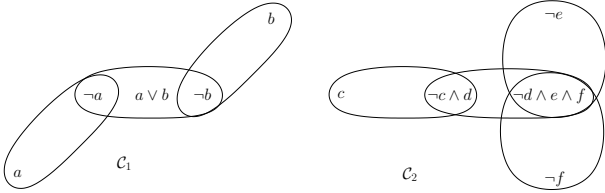


Fig 2: Connected components of K

Definition 13 (Maximal partial MUS-decomposition). A partial MUS-decomposition $T \in pMUSd(K)$ is called maximal if $|T| = \mu_D(K)$, where $\mu_D(K)$ is defined by

$$\mu_D(K) = \arg \max_{T' \in pMUSd(K)} |T'|.$$

Moreover, $\mu_D(K)$ is called the distribution index of K .

That is, the maximal partial MUS-decomposition has the largest cardinality among all partial MUS-decompositions. And the distribution index is the cardinality of maximal MUS-decompositions.

Example 6. (Example 5 contd.) Among maximal partial MUS-decompositions is $\{K'_1, K'_2, K'_3\}$. Note that C_2 contains highly connected formulae that cannot be separated into a partial MUS-decomposition of size larger than 2.

Although a (maximal) partial MUS-decomposition can be formed by any subsets of K , the next proposition indicates that only $MUSes(K)$ are needed to obtain a (maximal) partial MUS-decomposition.

Lemma 8. Let K be an inconsistent KB. There exist $\mu_D(K)$ distinct $MUSes$ $M_1, \dots, M_{\mu_D(K)}$ such that $\{M_i \mid 1 \leq i \leq \mu_D(K)\}$ is a maximal partial MUS-decomposition of K .

Proof. Suppose $\{K_1, \dots, K_{\mu_D(K)}\}$ is a maximal MUS-decomposition of K . Let $M_i \in MUSes(K_i)$ for $1 \leq i \leq \mu_D(K)$, then it is easy to verify that $\{M_1, \dots, M_{\mu_D(K)}\}$ is a partial MUS-decomposition whose cardinality is the distribution index of K , so it is a maximal pMUSd. \square

That is, each element of a maximal partial MUS-decomposition can be some minimal unsatisfiable subsets of K , as $\{K'_1, K'_2, K'_3\}$ in Example 5. Moreover, the following proposition tells that we can have another special format of maximal MUS-decomposition.

Proposition 9 (Distributable MUS-decomposition). Let K be an inconsistent KB. There exist $\mu_D(K)$ distinct $M_i \subseteq MUSes(K)$ for $1 \leq i \leq \mu_D(K)$, such that $\{\bigcup_{M \in \mathcal{M}_i} M \mid 1 \leq i \leq \mu_D(K)\}$ is a maximal partial MUS-decomposition of K and \mathcal{M}_i is maximal w.r.t. set inclusion. We call such a maximal partial MUS-decomposition a distributable MUS-decomposition.

Proof. By Lemma 8, take a maximal MUS-decomposition of the form $\{M_i \in MUSes(K) \mid 1 \leq i \leq \mu_D(K)\}$. Denote C_i the connected component of $\mathcal{G}_{MUS}(K)$ such that $M_i \in C_i$. Now consider $\mathcal{M}_i \subseteq C_i$ such that $\{\bigcup_{M \in \mathcal{M}_i} M \mid 1 \leq i \leq \mu_D(K)\}$ is still a partial MUS-decomposition of K . Such \mathcal{M}_i exists because we can take $\mathcal{M}_i = \{M_i\}$. K is finite, so are $\mathcal{G}_{MUS}(K)$ and C_i . Now taking \mathcal{M}_i that is maximal w.r.t. set-inclusion with such a property, the conclusion follows. \square

Example 7. (Example 5 contd.) $\{K'_1, K'_2, C_2\}$ is a distributable MUS-decomposition, but $\{K'_1, K'_2, K'_3\}$ is not because $K'_3 \subset C_2$.

Example 8. Recall the example in Section 1: $K = \{a_1, \neg a_1, a_1 \vee \neg a_2, a_2, \neg a_2, \dots, a_{n-1}, \neg a_{n-1}, a_{n-1} \vee \neg a_n, a_n, \neg a_n\}$. The distributable MUS-decomposition of K is $\{a_i, \neg a_i\}$.

A distributable MUS-decomposition defines a way to separate a whole KB into maximal number of disjoint inconsistent components. The decomposed components, such as $\{a_i, \neg a_i\}$, can in turn be delivered to n different experts to repair in parallel. In the case where resolving inconsistency is a serious and time-consuming decision, this can advance task time by a distributed manipulation of maximal experts. Indeed, the rational in distribution MUS-decomposition related to inconsistency resolving is given in the following proposition.

Proposition 10. Given an inconsistent base K and $T = \{K_1, \dots, K_n\}$ is a distributable MUS-decomposition of K . Suppose K'_i is a consistent base obtained by removing or weakening formulae in K_i . Then $K' = \bigcup_1^n K'_i$ is consistent.

That is, inconsistencies in each component can be resolved separately and the merged KB afterwards is consistent. However, note that $K' \cup R$ where $R = K \setminus \bigcup_i K_i$ is not necessarily consistent⁶. For instance, in Example 6, if we have $K'_1 = \{\neg a\}$ and $K'_2 = \{\neg b\}$ after expert verification,

⁶Indeed, this is unavoidable by Proposition 13 if each expert only removes one formula from K .

we still have inconsistency in $\{\neg a, a \vee b, \neg b\}$. In this case, we can drop $a \vee b$ because $\{\neg a, \neg b\}$ have been manually chosen by experts; Or for carefulness, we can retrigger the same process to resolve the rest inconsistencies.

5.2 Distribution-based Inconsistency Degree

As we can see above that a distributable MUS-decomposition gives a reasonable disjoint partition of a KB. In this section, we study the distribution index which rises an interesting inconsistency measure with desired properties.

Definition 14. Let K be a KB, the distribution-based inconsistency degree $I_{\mathcal{D}}(K)$ is defined as:

$$I_{\mathcal{D}}(K) = \mu_{\mathcal{D}}(K).$$

Intuitively, $I_{\mathcal{D}}(K)$ characterizes how many experts are demanded to repair inconsistencies in parallel. The higher the value is, more labor force is required⁷.

Example 9. (Example 7 contd.) Since $\{K_1', K_2', C_2\}$ is a distributable MUS-decomposition, we have $I_{\mathcal{D}}(K) = 3$.

Indeed, the so defined measure satisfies several important properties for an inconsistency measure.

Proposition 11. $I_{\mathcal{D}}(K)$ satisfies Consistency, Monotony, Free formula independence, MinInc, and Independent Decomposability.

Proof. Consistency: If K is consistent, the partial MUS-decomposition set is empty, so $I_{\mathcal{D}}(K) = 0$.

Monotony: For any KB K and K' , it is easy to see that a partial MUS-decomposition of K is a partial MUS-decomposition of $K \cup K'$. Therefore, $\mu_{\mathcal{D}}(K) \leq \mu_{\mathcal{D}}(K \cup K')$.

Free formula independence: It follows from the obvious fact that free formula do not effect the set of partial MUS-decompositions.

MinInc: For $M \in MUSes(K)$, clearly, the only partial decomposition of M is $\{M\}$, so $I_{\mathcal{D}}(K) = 1$.

Independent Decomposability: Let K, K' two bases satisfying $MUSes(K) \oplus MUSes(K') = MUSes(K \cup K')$ and $unfree(K) \cap unfree(K') = \emptyset$. For any partial MUS-decompositions of K and K' : $\mathcal{M} = \{M_1, \dots, M_{\mu_{\mathcal{D}}(K)}\}$ and $\mathcal{M}' = \{M'_1, \dots, M'_{\mu_{\mathcal{D}}(K')}\}$, it is easy to see $\mathcal{M} \cup \mathcal{M}' \in pMUSd(K \cup K')$. Moreover, $\mathcal{M} \cup \mathcal{M}'$ is of the maximal cardinality in $pMUSd(K \cup K')$. Otherwise, by Lemma 8, there are $M''_j \in MUSes(K \cup K')$ that form a partial MUS-decomposition of $K \cup K'$: $\{M''_1, \dots, M''_N\}$ with $N > \mu_{\mathcal{D}}(K) + \mu_{\mathcal{D}}(K')$. Since $MUSes(K) \oplus MUSes(K') = MUSes(K \cup K')$, we have either $M''_j \in MUSes(K)$ or $M''_j \in MUSes(K')$ for all j . So at least one of K and K' has a partial MUS-decomposition whose cardinality is stricter larger than its distribution index. A contradiction with the definition of distribution index. So $\mu_{\mathcal{D}}(K \cup K') = \mu_{\mathcal{D}}(K) + \mu_{\mathcal{D}}(K')$. Consequently, $I_{\mathcal{D}}$ satisfies independent decomposability property. \square

⁷Note that labor force for resolving an inconsistent component can vary for different inconsistent components because of for instance differences in their sizes. But we focus on the study of maximal number of components in the present paper.

Moreover, the distribution-based inconsistency measure is a lower bound of inconsistency measures which satisfy monotony, independent Decomposability, and MinInc properties.

Proposition 12. Given an inconsistency measure I that satisfies Monotony, Independent Decomposability, and MinInc, we have $I(K) \geq \mu_{\mathcal{D}}(K)$.

Proof. For any partial MUS-decomposition $\{K_1, \dots, K_n\}$ of K , we have $\bigcup_{1 \leq i \leq n} K_i \subseteq K$. So by monotony, $I(K) \geq I(K_1 \cup \dots \cup K_n)$. Moreover, since I satisfies independent Decomposability, $I(K) \geq I(K_1) + \dots + I(K_n)$. Taking a maximal partial MUS-decomposition, one can deduce that $I(K) \geq I(K_1) + \dots + I(K_{\mu_{max}(K)})$. By MinInc and monotony, $I(K_i) \geq 1$, so $I(K) \geq \mu_{\mathcal{D}}(K)$. \square

Example 10. (Example 1 contd.) For different measures based on MUSes, we have

- $I_{\mathcal{D}}(K_1 \cup K_2) = 1$ and $I_{\mathcal{D}}(K_1 \cup K_3) = 2$;
- $\delta_{hs}(K_1 \cup K_2) = 1$ and $\delta_{hs}(K_1 \cup K_3) = 2$;
- $I'_M(K_1 \cup K_2) = 1$ and $I'_M(K_1 \cup K_3) = 4$;
- $I_{MI}(K_1 \cup K_2) = 2$ and $I_{MI}(K_1 \cup K_3) = 2$.

So all $I_{\mathcal{D}}, \delta_{hs}$, and I'_M give a conclusion that $K_1 \cup K_2$ is less inconsistent than $K_1 \cup K_3$, which coincides with our intuition, but it is not the case of I_{MI} .

In Example 10, we have $I_{\mathcal{D}}$ and δ_{hs} of the same value. But it is not the general case as shown in the following example.

Example 11. (Example 5 contd.) For the connected component C_2 , $\delta_{hs}(C_2) = 2$ while its distribution index is 1.

However, the following Proposition gives a general relationship between $I_{\mathcal{D}}$ and δ_{hs} .

Proposition 13. Let K be a KB. We have

$$I_{\mathcal{D}}(K) \leq \delta_{hs}(K).$$

Proof. As K can be partitioned into $\mu_{\mathcal{D}}(K)$ disjoint components of minimal inconsistent subsets of K , a minimal hitting set of K must contain at least one formula from each component. That is, $I_{\mathcal{D}}(K) = \mu_{\mathcal{D}}(K) \leq \delta_{hs}(K)$. \square

Example 12. (Example 8 contd.) We have $I_{\mathcal{D}}(K) = n$ and $\delta_{hs}(K) = n$. But the former means that K can be distributed to n experts to resolve inconsistency in parallel and each expert only verifies two elements because of the distribution MUS-decomposition is $\{a_i, \neg a_i\}$; Whilst the latter means that each expert needs to verify at least n formulae to confirm an inconsistency resolving plan. And different experts have to do repetition work due to overlapping among different hitting sets.

This example shows that the proposed MUS-decomposition gives a more competitive inconsistency handling methodology than the hitting set based approach albeit the occasionally equivalent value of the deduced inconsistency measures $I_{\mathcal{D}}(K)$ and δ_{hs} .

6 Computations of $I_{\mathcal{D}}(K)$

In this section, we consider the computational issues of distribution-based inconsistency measure $I_{\mathcal{D}}(K)$ by generalizing the classical Set Packing problem, and then show two encodings of $I_{\mathcal{D}}(K)$, which is aiming at practical algorithms for its solution.

We first look at the following proposition which is a simple conclusion of Lemma 8.

Proposition 14. *Let K be a KB. $I_{\mathcal{D}}$ is the maximal cardinality of $\mathcal{M} \subseteq \text{MUSes}(K)$ satisfying*

1. $\text{MUSes}(\cup_{M \in \mathcal{M}} M) = \mathcal{M}$.

Proposition 14 states that $I_{\mathcal{D}}(K)$ is the largest number of (pairwise disjoint) MUSes of K such that their union will not rise any new MUS, which gives a way to compute $I_{\mathcal{D}}(K)$.

Next we study this computation in the framework of Maximum Closed Set Packing (MCSP) defined in the following.

6.1 Closed Set Packing

The maximum set packing problem is one of the basic optimization problems (see, e.g., (Garey and Johnson 1990)). It is related to other well-known optimization problems, such as the maximum independent set and maximum clique problems (Arora et al. 1998; Arora and Safra 1998; Boppana and Halldórsson 1992; Feige et al. 1996; Wigderson 1983). We here introduce a variant of this problem, called the *maximum closed set packing problem*. We show that this variant is NP-hard by providing a reduction from the maximum set packing problem which is NP-hard (Karp 1972). In this work, the maximum closed set packing problem is used to compute the distribution-based inconsistency measure.

Let U be universe and S be a family of subsets of U .

Definition 15 (Set Packing). *A set packing is a subset $P \subseteq S$ such that, for all $S_i, S_j \in P$ with $S_i \neq S_j$, $S_i \cap S_j = \emptyset$.*

Our variant is obtained from the maximum set packing problem by further requiring that the union of selected subsets does not contain unselected subsets in S as defined below.

Definition 16 (Closed Set Packing). *A closed set packing is a set packing $P \subseteq S$ such that, for all $S_i \in S \setminus P$, S_i is not a subset of $\bigcup_{P_i \in P} P_i$.*

The *maximum (free) set packing problem* consists in founding a (free) set packing with maximum cardinality, written MSP (MCSP).

Theorem 15. *MCSP is NP-hard.*

Proof. We construct a reduction from the maximum set packing problem to the maximum closed set packing problem. Let U be a universe, $S = \{S_1, \dots, S_n\}$ a family of subsets of U and e_1, \dots, e_n are n distinct elements which do not belong to U . Define $U' = U \cup \{e_1, \dots, e_n\}$ and $S' = \{S_1 \cup \{e_1\}, \dots, S_n \cup \{e_n\}\}$. We have P is a solution of the maximum set packing problem for (U, S) if and only if $P' = \{S_i \cup \{e_i\} \mid S_i \in P\}$ is a solution of the maximum closed set packing problem for (U', S') . Since maximum set packing MSP is NP-hard, so is the MCSP. \square

6.2 Integer Linear Program Formulation of MCSP

We here provide an encoding of the maximum closed set packing problem in linear integer programming. Let U be a universe and S a set of subsets of U . We associate a binary variable X_{S_i} ($X_{S_i} \in \{0, 1\}$) to each subset S_i in S . We also associate a binary variable Y_e to each element e in U .

The first linear inequalities allow us to only consider the pairwise disjoint subsets in S :

$$\sum_{e \in S_i, S_j \in S} X_{S_i} \leq 1 \quad \text{for all } e \in U \quad (1)$$

The following inequalities allow us to have $X_{S_i} = 1$ if and only if, for all $e \in S_i$, $Y_e = 1$:

$$\left(\sum_{e \in S_i} Y_e\right) - C_i * X_{S_i} \geq 0 \quad \text{for all } S_i \in S \quad (2)$$

$$\left(\sum_{e \in S_i} Y_e\right) - X_{S_i} \leq C_i - 1 \quad \text{for all } S_i \in S \quad (3)$$

where, for all $S_i \in S$, $C_i = |S_i|$. Indeed, If $X_{S_i} = 1$ then, using inequality (2), we have, for all $e \in S_i$, $Y_e = 1$. Otherwise, we have $X_{S_i} = 0$ and, using inequality (3), there exists $e \in S_i$ such that $Y_e = 0$.

Finally, the objective function is defined as follows:

$$\max \sum_{S_i \in S} X_{S_i} \quad (4)$$

Proposition 16. *The linear inequalities in (1), (2) and (3) with the objective function (4) is a correct encoding of MCSP.*

Proof. Let P be a subset of S that corresponds to a solution of the linear integer program. Using the inequalities in (1), we have, for all $S_i, S_j \in P$ with $S_i \neq S_j$, $S_i \cap S_j = \emptyset$. Thus, P corresponds to a set packing. Using the inequalities (2) and (3), we have, for all $S_i \in S$, $X_{S_i} = 1$ if and only if, for all $e \in S_i$, $Y_e = 1$. Hence, for all $S_i \in S \setminus P$, there exists $e \in S_i$ such that $Y_e = 0$, so S_i is not a subset of $\bigcup_{P_i \in P} P_i$. Therefore, P is a closed set packing. Finally, from maximizing the objective function in (4), we deduce that P is a solution of the maximum closed set packing for (U, S) . \square

6.3 MinCostSAT Formulation of MCSP

In this section, we describe our encoding of the maximum closed set packing problem as a MinCostSAT instance (Miyazaki, Iwama, and Kambayashi 1996).

Definition 17 (MinCostSAT). *Let Φ be a CNF formula and f a cost function that associates a non-negative cost to each variable in $\text{Var}(\Phi)$. The MinCostSAT problem is the problem of finding a model for Φ that minimizes the objective function:*

$$\sum_{p \in \text{Var}(\Phi)} f(p)$$

Let U be a universe and S a set of subsets of U . We associate a boolean variable X_{S_i} (resp. Y_e) to each $S_i \in S$ (resp. $e \in U$). The inequalities in (1) in our previous integer linear program correspond to instances of the AtMostOne constraint which is a special case of the well-known cardinality constraint. Several efficient encodings of the cardinality constraint to CNF have been proposed, most of them try to improve the efficiency of constraint propagation (e.g. (Bailleux and Boufkhad 2003; Sinz 2005)). We here consider the encoding using sequential counter (Sinz 2005; Silva and Lynce 2007). In this case, the inequality $\sum_{e \in S_i, S_i \in S} X_{S_i} \leq 1$ is encoded as follows (we fix $\sum_{e \in S_i, S_i \in S} X_{S_i} = \sum_{1 \leq i \leq n} X_{S_i}$):

$$\begin{aligned} & (\neg X_{S_1} \vee p_1) \wedge (\neg X_{S_n} \vee \neg p_{n-1}) \\ & \bigwedge_{1 < i < n} ((\neg X_{S_i} \vee p_i) \wedge (\neg p_{i-1} \vee p_i) \wedge (\neg X_{S_i} \vee \neg p_{i-1})) \end{aligned} \quad (5)$$

where p_i is a fresh boolean variable for all $1 \leq i \leq n-1$.

Regarding to the inequalities in (2), it can be encoded by the following clauses:

$$\bigwedge_{S_i \in S} \bigwedge_{e \in S_i} \neg X_{S_i} \vee Y_e \quad (6)$$

Indeed, these clauses are equivalent to the following ones:

$$\bigwedge_{S_i \in S} X_{S_i} \rightarrow \bigwedge_{e \in S_i} Y_e$$

The inequalities in (3) can be simply encoded as:

$$\bigwedge_{S_i \in S} (X_{S_i} \vee \bigvee_{e \in S_i} \neg Y_e) \quad (7)$$

Contrary to MCSP, the optimization process in MinCostSAT consists in minimizing the objective function. In order to encode MCSP as an MinCostSAT instance, we rename each variable X_{S_i} with $\neg X'_{S_i}$ (X'_{S_i} is a fresh boolean variable) in (5), (6) and (7), for all $S_i \in S$. The MinCostSAT instance encoding the maximum closed set packing problem for (U, S) is $\mathcal{M}_{(U, S)} = (\Phi, f)$ where Φ is the CNF formula obtained from (5) \wedge (6) \wedge (7) by the renaming described previously and f is defined as follows:

- for all $S_i \in S$, $f(X'_{S_i}) = 1$; and
- for all $v \in \text{Var}(\Phi) \setminus \{X'_{S_i} \mid S_i \in S\}$, $f(v) = 0$.

Note that the optimization process in \mathcal{M} consists in minimizing $\sum_{S_i \in S} X'_{S_i}$ and that corresponds to maximizing $\sum_{S_i \in S} X_{S_i}$.

7 Experimental Results

In this section, we present a preliminary experimental evaluation of our proposed approach. All experiments was performed on a Xeon 3.2GHz (2 GB RAM) cluster.

We conducted two kinds of experiments. The first one deals with instances coming from classical MUSes enumeration problem. For this category we use two complementary

state-of-the art MUSes enumeration solvers and then we apply our encoding into MCSP to compute the values of I_D . When enumerating all MUSes is infeasible we use eMUS (Previti and Marques-Silva 2013) instead of camus (Liffiton and Sakallah 2008) to enumerate a subset of MUSes. Indeed, eMUS is a real time solver that outperforms camus when we deal with partial MUSes enumeration. The instances where eMUS is used are indicated with an asterisk.

In the second experiment, the instances are randomly generated. To represent a KB with n formulae involving m MUSes, called mfsp-m-n, we first generate randomly a family of sets $\{S_1, \dots, S_m\}$ of positive integers from the interval $[1 \dots n]$. We suppose that each set S_i of numbers represents a MUS. We randomly set the size of S_i . In our experiments, we consider $1 < |S_i| \leq 3$.

In Table 1, for each instance, we report the number of MUSes ($\#mus$), the value of the inconsistency measure (I_D) and the time ($time$ in seconds) needed to compute I_D . To solve the encoded instances, we use wpmxatz Partial MaxSAT solver (Argelich et al. 2006).

As we can observe, the value I_D is much smaller than the number of MUSes. Furthermore, the computation time globally increases as I_D increases. Note that for instances whose I_D value is equal to 1, it means that they are strongly interconnected.

Instance	#mus	I_D	time(s)
C168_FW_UT.851	102	1	1
C220_FV_RZ.13	6772	1	5.4
c880_gr_rcs.w5.shufed	70	1	3.7
rocket_ext.b	75	1	1
c7552-bug-gate-0*	1000	1	5.3
apex_gr_2pin.w4.shuffled*	1500	2	120.23
wb_conmax1.dimacs.filtered*	20	2	0.9
wb_4m8s4.dimacs.filtered*	20	9	1.44
mfsp_50_20	50	5	0.01
mfsp_100_50	100	22	0.36
mfsp_120_60	120	15	1.49
mfsp_120_80	120	20	13.78
mfsp_150_60	150	11	1.50
mfsp_150_100	150	22	127.57
mfsp_150_150	150	35	347.98
mfsp_200_50	200	11	4.79

Table 1: Computation of I_D (real-world and random instances)

8 Conclusion

We studied in this paper a new framework for characterizing inconsistency based on the proposed independent decomposability property and MUS-decomposition. Such defined inconsistency measures (i.e. I'_M and I_D) are shown with desired properties. The distributable MUS-decomposition allows to resolve inconsistencies in a parallel way, which is a rarely considered methodology for handling large knowledge bases with important informations. Complexity and practical algorithms are studied based on the advance of MUS enumeration. We will study the lower bound complexity of the measure and explore applications of the proposed

methodology in the future.

References

- [Argelich et al. 2006] Argelich, J.; Domingo, X.; Li, C. M.; Manyà, F.; and Planes, J. 2006. Towards solving many-valued maxsat. In *ISMVL*, 26.
- [Arora and Safra 1998] Arora, S., and Safra, S. 1998. Probabilistic checking of proofs: A new characterization of np. *J. ACM* 45(1):70–122.
- [Arora et al. 1998] Arora, S.; Lund, C.; Motwani, R.; Sudan, M.; and Szegedy, M. 1998. Proof verification and the hardness of approximation problems. *J. ACM* 45(3):501–555.
- [Bailleux and Boufkhad 2003] Bailleux, O., and Boufkhad, Y. 2003. Efficient cnf encoding of boolean cardinality constraints. In *9th International Conference on Principles and Practice of Constraint Programming - CP 2003*, 108–122.
- [Bertossi, Hunter, and Schaub 2005] Bertossi, L. E.; Hunter, A.; and Schaub, T. 2005. Introduction to inconsistency tolerance. In *Inconsistency Tolerance*, 1–14.
- [Boppana and Halldórsson 1992] Boppana, R. B., and Halldórsson, M. M. 1992. Approximating maximum independent sets by excluding subgraphs. *BIT* 32(2).
- [Chen, Zhang, and Zhang 2004] Chen, Q.; Zhang, C.; and Zhang, S. 2004. A verification model for electronic transaction protocols. In *APWeb*, 824–833.
- [Doder et al. 2010] Doder, D.; Raskovic, M.; Markovic, Z.; and Ognjanovic, Z. 2010. Measures of inconsistency and defaults. *Int. J. Approx. Reasoning* 51(7):832–845.
- [Feige et al. 1996] Feige, U.; Goldwasser, S.; Lovász, L.; Safra, S.; and Szegedy, M. 1996. Interactive proofs and the hardness of approximating cliques. *J. ACM* 43(2):268–292.
- [Garey and Johnson 1990] Garey, M. R., and Johnson, D. S. 1990. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. New York, NY, USA: W. H. Freeman Co.
- [Grant and Hunter 2006] Grant, J., and Hunter, A. 2006. Measuring inconsistency in knowledgebases. *J. Intell. Inf. Syst.* 27(2):159–184.
- [Grant and Hunter 2008] Grant, J., and Hunter, A. 2008. Analysing inconsistent first-order knowledgebases. *Artif. Intell.* 172(8-9):1064–1093.
- [Grant and Hunter 2011] Grant, J., and Hunter, A. 2011. Measuring consistency gain and information loss in stepwise inconsistency resolution. In *ECSQARU*, 362–373.
- [Grant and Hunter 2013] Grant, J., and Hunter, A. 2013. Distance-based measures of inconsistency. In *ECSQARU*, 230–241.
- [Grant 1978] Grant, J. 1978. Classifications for inconsistent theories. *Notre Dame Journal of Formal Logic* 19(3):435–444.
- [Hunter and Konieczny 2006a] Hunter, A., and Konieczny, S. 2006a. Shapley inconsistency values. In *KR*, 249–259.
- [Hunter and Konieczny 2006b] Hunter, A., and Konieczny, S. 2006b. Shapley inconsistency values. In *KR*, 249–259.
- [Hunter and Konieczny 2008] Hunter, A., and Konieczny, S. 2008. Measuring inconsistency through minimal inconsistent sets. In *KR*, 358–366.
- [Hunter and Konieczny 2010] Hunter, A., and Konieczny, S. 2010. On the measure of conflicts: Shapley inconsistency values. *Artif. Intell.* 174(14):1007–1026.
- [Hunter 2002] Hunter, A. 2002. Measuring inconsistency in knowledge via quasi-classical models. In *AAAI/IAAI*, 68–73.
- [Hunter 2006] Hunter, A. 2006. How to act on inconsistent news: Ignore, resolve, or reject. *Data Knowl. Eng.* 57(3):221–239.
- [Jabbour and Raddaoui 2013] Jabbour, S., and Raddaoui, B. 2013. Measuring inconsistency through minimal proofs. In *ECSQARU*, 290–301.
- [Karp 1972] Karp, R. M. 1972. Reducibility among combinatorial problems. In *Complexity of Computer Computations*, 85–103. Plenum Press, New York.
- [Knight 2002] Knight, K. 2002. Measuring inconsistency. *J. Philosophical Logic* 31(1):77–98.
- [Liffiton and Sakallah 2008] Liffiton, M. H., and Sakallah, K. A. 2008. Algorithms for computing minimal unsatisfiable subsets of constraints. *J. Autom. Reasoning* 40(1):1–33.
- [Luce and Raiffa 1957] Luce, R. D., and Raiffa, H. 1957. *Games and Decision*. Wiley.
- [Ma et al. 2010] Ma, Y.; Qi, G.; Xiao, G.; Hitzler, P.; and Lin, Z. 2010. Computational complexity and anytime algorithm for inconsistency measurement. *Int. J. Software and Informatics* 4(1):3–21.
- [Ma, Qi, and Hitzler 2011] Ma, Y.; Qi, G.; and Hitzler, P. 2011. Computing inconsistency measure based on paraconsistent semantics. *J. Log. Comput.* 21(6):1257–1281.
- [Martinez, Arias, and and 2004] Martinez, A. B. B.; Arias, J. J. P.; and and, A. F. V. 2004. On measuring levels of inconsistency in multi-perspective requirements specifications. In *PRISE’04*, 21–30.
- [Martinez et al. 2007] Martinez, M. V.; Pugliese, A.; Simari, G. I.; Subrahmanian, V. S.; and Prade, H. 2007. How dirty is your relational database? an axiomatic approach. In *ECSQARU*, 103–114.
- [McAreavey et al. 2011] McAreavey, K.; Liu, W.; Miller, P.; and Mu, K. 2011. Measuring inconsistency in a network intrusion detection rule set based on snort. *Int. J. Semantic Computing* 5(3).
- [Miyazaki, Iwama, and Kambayashi 1996] Miyazaki, S.; Iwama, K.; and Kambayashi, Y. 1996. Database queries as combinatorial optimization problems. In *CODAS*, 477–483.
- [Mu et al. 2011] Mu, K.; Liu, W.; Jin, Z.; and Bell, D. A. 2011. A syntax-based approach to measuring the degree of inconsistency for belief bases. *Int. J. Approx. Reasoning* 52(7):978–999.
- [Mu, Liu, and Jin 2011] Mu, K.; Liu, W.; and Jin, Z. 2011. A general framework for measuring inconsistency through minimal inconsistent sets. *Knowl. Inf. Syst.* 27(1):85–114.
- [Mu, Liu, and Jin 2012] Mu, K.; Liu, W.; and Jin, Z. 2012.

- Measuring the blame of each formula for inconsistent prioritized knowledge bases. *J. Log. Comput.* 22(3):481–516.
- [Oller 2004] Oller, C. A. 2004. Measuring coherence using lp-models. *J. Applied Logic* 2(4):451–455.
- [Previti and Marques-Silva 2013] Previti, A., and Marques-Silva, J. 2013. Partial mus enumeration. In *AAAI*.
- [Qi, Liu, and Bell 2005] Qi, G.; Liu, W.; and Bell, D. A. 2005. Measuring conflict and agreement between two prioritized belief bases. In *IJCAI*, 552–557.
- [Shapley 1953] Shapley, L. S. 1953. A value for n-person games. In *H. W. Kuhn and A. W. Tucker, editors, Contributions to the Theory of Games II (Annals of Mathematics Studies 28)*, 307–317.
- [Silva and Lynce 2007] Silva, J. P. M., and Lynce, I. 2007. Towards robust cnf encodings of cardinality constraints. In *CP*, 483–497.
- [Sinz 2005] Sinz, C. 2005. Towards an optimal cnf encoding of boolean cardinality constraints. In *11th International Conference on Principles and Practice of Constraint Programming - CP 2005*, 827–831.
- [Wigderson 1983] Wigderson, A. 1983. Improving the performance guarantee for approximate graph coloring. *J. ACM* 30(4):729–735.
- [Xiao and Ma 2012] Xiao, G., and Ma, Y. 2012. Inconsistency measurement based on variables in minimal unsatisfiable subsets. In *ECAI*, 864–869.
- [Xiao et al. 2010] Xiao, G.; Lin, Z.; Ma, Y.; and Qi, G. 2010. Computing inconsistency measurements under multi-valued semantics by partial max-sat solvers. In *KR*.
- [Zhou et al. 2009] Zhou, L.; Huang, H.; Qi, G.; Ma, Y.; Huang, Z.; and Qu, Y. 2009. Measuring inconsistency in dl-lite ontologies. In *Web Intelligence*, 349–356.